Scaling Up Approximate Value Iteration with Options: Better Policies with Fewer Iterations

Timothy A. Mann MANN@EE.TECHNION.AC.IL
Shie Mannor SHIE@EE.TECHNION.AC.IL
Department of Electrical Engineering, The Technion - Israel Institute of Technology, Haifa, Israel 32000

Abstract

We show how options, a class of control structures encompassing primitive and temporally extended actions, can play a valuable role in planning in MDPs with continuous state-spaces. Analyzing the convergence rate of Approximate Value Iteration with options reveals that for pessimistic initial value function estimates, options can speed up convergence compared to planning with only primitive actions even when the temporally extended actions are suboptimal and sparsely scattered throughout the state-space. Our experimental results in an optimal replacement task and a complex inventory management task demonstrate the potential for options to speed up convergence in practice. We show that options induce faster convergence to the optimal value function, which implies deriving better policies with fewer iterations.

1. Introduction

Under most analyses of approximate dynamic programming, one iteration corresponds to planning one additional timestep into the future. On the other hand, by implementing Approximate Value Iteration (AVI) with temporally extended actions, one iteration could instead correspond to planning several timesteps into the future. We derive bounds that help us reason about when AVI with temporally extended actions converges faster than AVI with only primitive actions and also how much faster the convergence may be.

Previous studies have considered planning with options or temporally extended actions. Precup et al. (1998) demonstrated that value iteration and policy iteration converge when planning with options in Markov Decision Processes (MDPs, defined in section 2). Sutton et al. (1999) show experimental results in a discrete state MDP where options speed up the convergence rate of planning, but the theoretical results of Precup et al. (1998) do not specify any clear advantages of planning with options compared to planning with only primitive actions. Silver & Ciosek (2012) demonstrate impressive results for planning while simultaneously composing new options in multiple discrete state MDPs. Silver & Ciosek further show that the options generated by their proposed algorithm converge to optimal, but their theoretical analysis does not compare convergence rates. Hauskrecht et al. (1998) showed (for discrete state MDPs) that when the initial value function estimate V_0 is pessimistic, then planning with additional options converges faster than planning with only primitive actions. However, the analysis of Hauskrecht does not characterize how much faster planning with options may be compared to planning with only primitive actions. In partially observable MDPs, Theocharous & Kaelbling (2003) and He et al. (2011) used sequences of actions, called macro-actions, to speed up tree-based search for near-optimal actions. He et al. (2011) found that planning with macro-actions could help to ease (but not eliminate) the exponential dependence on the horizon time inherent to tree-based search methods. While we cannot make a direct comparison to our setting, one advantage of our analysis is that the speed up achieved by planning with options does not depend on explicitly eliminating primitive actions from consideration. On the other hand, Theocharous & Kaelbling (2003) and He et al. (2011) achieve faster performance by explicitly eliminating some primitive actions from consideration. This can have the negative side affect of convergence toward suboptimal policies.

The options framework is appealing for investigating planning with temporally extended actions. For one thing, the class of options includes both primitive actions and a range of temporally extended actions. Many of the well-known properties of Markov decision processes generalize when arbitrary options are added (e.g., Value Iteration and Policy Iteration still converge (Precup et al., 1998; Sutton et al.,

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

1999)). In addition, much effort has gone into algorithms that learn "good" options for exploration (Stolle & Precup, 2002; Mannor et al., 2004). These algorithms may produce options that are also useful for planning. Lastly, options enable greater flexibility to model real-world problems where the time between decisions may vary. For example, in inventory management problems, orders may be placed when inventory is running low. This strategy makes the time between orders a random variable, which is more naturally modeled by options than primitive actions. Thus, options are an important candidate for investigating planning with temporally extended actions.

The main technical contributions of this paper are extending the finite-sample/finite-iteration analysis of AVI to planning with options. First, we introduce a generalization of the Fitted Value Iteration (FVI) algorithm that incorporates samples generated by options. We show that when the set of options contains the primitive actions, our generalized algorithm converges approximately as fast as FVI with only primitive actions (Proposition 1). Then we develop precise conditions under which our generalized FVI algorithm converges faster with options than with only primitive actions (Theorem 1). These conditions turn out to depend critically on whether the iterates produced by FVI underestimate the optimal value function. Finally, our experimental results in two domains demonstrate that the convergence rate of planning with options can be significantly faster than planning with only primitive actions. However, as predicted by our theoretical analysis, the improvement in convergence only occurs when the iterates of our algorithm underestimate the optimal value function, which can be controlled in practice by setting the initial estimate of the optimal value function pessimistically. Our analysis of FVI suggests that options can play an important role in planning by inducing fast convergence.

2. Background

A Markov Decision Process (MDP) is defined by a 5-tuple $\langle X, A, P, R, \gamma \rangle$ where X is a set of states, A is a set of primitive actions, P maps from state-action pairs to a probability distribution over states, R is a mapping from state-action pairs to reward distributions bound to the interval $[-R_{\max}, R_{\max}]$, and $\gamma \in [0, 1)$ is a discount factor. Let $B(X; V_{\max})$ denote the set of functions with domain X and range bounded by $[-V_{\max}, V_{\max}]$ and $M(X)$ the set of all probability measures on X. Throughout this paper we will consider MDPs where X is a bounded subset of a d-dimensional Euclidean space and A is a finite set of primitive actions.

A deterministic, stationary policy $\pi : X \to A$ for an MDP is a mapping from states to primitive actions. We denote the set of deterministic, stationary policies by Π. The objective of planning in an MDP is to derive a policy π that maximizes $V^\pi(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t(x_t, \pi(x_t)) | x_0 = x, \pi\right]$, where x is the long-term value of following π starting in state x. The function V^π is called the value function of the policy π and it is well known that it can be written recursively as $\mathcal{T}^\pi V^\pi = \mathbb{E}\left[R(x, \pi(x))\right] + \gamma \int P(y|x, \pi(x)) V^\pi(y) dy$, where $\mathcal{T}^\pi$ is a backup operator with respect to π and V^π is its unique fixed point. Given $V \in B(X; V_{\max})$, the greedy policy π with respect to V is defined by $\pi(x) = \arg\max_{a \in A} \mathbb{E}\left[R(x, a)\right] + \gamma \int P(y|x) V(y) dy$. We denote the optimal value function by $V^* = \max_{\pi \in \Pi} V^\pi$. A policy π^* is optimal if its corresponding value function is V^* and a policy π is α-optimal if $V^\pi(x) \geq V^*(x) - \alpha$ for all $x \in X$. The Bellman operator $\mathcal{T}$ is defined by

$$(\mathcal{T}V)(x) = \max_{a \in A} \left(\mathbb{E}\left[R(x, a)\right] + \gamma \int P(y|x, a) V(y) dy \right) , \quad (1)$$

where $V \in B(X; V_{\max})$, which is known to have fixed point V^*. Equation (1) defines the Value Iteration (VI) algorithm. VI converges to V^*, but it is computationally intractable in MDPs with extremely large or continuous state-spaces.

Primitive action Fitted Value Iteration (PFVI) is a generalization of VI to handle large or continuous state-spaces. PFVI runs iteratively producing a sequence of $K \geq 1$ estimates $\{V_k\}_{k=1}^K$ of the optimal value function and returns a policy π_K that is greedy with respect to the final estimate V_K. During each iteration k, the algorithm computes a set of empirical estimates $\hat{V}_k$ of $\mathcal{T}V_{k-1}$ for N states, and then fits a function approximator to $\hat{V}_k$. To generate $\hat{V}_k$, N states $\{x_i\}_{i=1}^N$ are sampled from a distribution $\mu \in M(X)$. For each sampled state x_i and each primitive action $a \in A$, L next states $\{y_{i,j}^a\}_{j=1}^L$ and rewards $\{r_{i,j}^a\}_{j=1}^L$ are sampled from the MDP simulator $\mathbb{S}$. For the k^{th} iteration, the estimates of the Bellman backups are computed by

$$\hat{V}_k(x_i) = \max_{a \in A} \frac{1}{L} \sum_{j=1}^{L} \left(r_{i,j}^a + \gamma V_{k-1}(y_{i,j}^a) \right) , \quad (2)$$

where V_0 is the initial estimate of the optimal value function given as an argument to PFVI. The k^{th} estimate of the optimal value function is obtained by applying a supervised learning algorithm, that produces

$$V_k = \arg\min_{f \in \mathcal{F}} \sum_{i=1}^{N} \left| f(x_i) - \hat{V}_k(x_i) \right|^p , \quad (3)$$

where $p \geq 1$ and $\mathcal{F} \subset B(X; V_{\max})$ is the hypothesis space of the supervised learning algorithm.

Munos & Szepesvári (2008) presented a finite-sample, finite-iteration analysis of PFVI with guarantees dependent on the L_p-norm rather than the more conservative max

norm. This enabled analysis of instances of PFVI that use one of the many supervised learning algorithms minimizing L_1 or L_2 norm. A key assumption needed for their analysis is the notion of discounted-average concentrability of future state distributions.

Assumption 1. $[A1(\rho,\mu)]$ *(Munos, 2005) Given two distributions $\rho,\mu \in M(X)$ and $m \geq 1$ arbitrary policies $\pi_1,\pi_2,\ldots,\pi_m$, we assume that $\rho P^{\pi_1}P^{\pi_2}\ldots P^{\pi_m}$ is absolutely continuous with respect to μ implying that*

$$c(m) \stackrel{def}{=} \sup_{\pi_1,\pi_2,\ldots,\pi_m} \left\| \frac{d(\rho P^{\pi_1}P^{\pi_2}\ldots P^{\pi_m})}{d\mu} \right\|_\infty < +\infty \ ,$$

and we assume that

$$C_{\rho,\mu} \stackrel{def}{=} (1-\gamma)^2 \sum_{m\geq 1} m\gamma^{m-1}c(m) < +\infty$$

is the discounted average concentrability coefficient.

Assumption 1 prevents too much transition probability mass from concentrating on a few states. The condition that $C_{\rho,\mu}$ is finite depends on $c(m)$ growing at most subexponentially. See Munos (2005) for a more complete discussion of Assumption 1.

Munos & Szepesvári (2008) showed that given an MDP, if we select probability distributions $\rho,\mu \in M(X)$, a positive integer p, a supervised learning algorithm over a bounded function space $\mathcal{F}$ satisfying (3), $V_0 \in \mathcal{F}$ an initial estimate of the optimal value function, and $\varepsilon > 0$ and $\delta \in (0,1]$. Then for any $K \geq 1$, with probability at least $1-\delta$, there exists parameters N and L such that the policy π_K returned by PFVI satisfies

$$\begin{aligned} ||V^* - V^{\pi_K}||_{p,\rho} &\leq \tfrac{2\gamma}{(1-\gamma)^2} C_{\rho,\mu}^{1/p} b_{p,\mu}(\mathcal{TF},\mathcal{F}) + \varepsilon \\ &+ \left(\gamma^{K+1}\right)^{1/p} \left(\tfrac{2\|V^*-V_0\|_\infty}{(1-\gamma)^2}\right) \ , \end{aligned} \tag{4}$$

where $\|f\|_{p,\rho} = \left(\int \rho(x)|f(x)|^p dx\right)^{1/p}$ and $b_{p,\mu}(\mathcal{TF},\mathcal{F}) = \sup_{f\in\mathcal{F}} \inf_{g\in\mathcal{F}} \|\mathcal{T}f - g\|_{p,\mu}$, the inherent Bellman error, is a measure of how well $\mathcal{F}$ represents $\hat{V}_k$ at each iteration. The first term in (4) controls the approximation error due to the fact that $\mathcal{F}$ does not exactly capture $\hat{V}_k$ at each iteration. The second term, the estimation error, is controlled by collecting more samples. This last term characterizes the convergence rate of the algorithm. It shrinks as K increases. The size of the discount factor γ controls the rate of convergence. Convergence is faster when γ is smaller. Unfortunately, γ is part of the problem definition. However, because options execute for multiple timesteps, an option can have an effective discount factor smaller than γ.

3. Semi-Markov Decision Processes

Semi-Markov Decision Processes (SMDPs) are a generalization of Markov Decision Processes (MDPs) that incorporates temporally extended actions. A set of primitive and temporally extended actions called options, denoted by $\mathcal{O}$, combined with an MDP forms an SMDP (Precup et al., 1998). An option o is defined by a 3-tuple $\langle \mathcal{I}_o, \pi_o, \beta_o \rangle$ where $\mathcal{I}_o$ is the set of states that o can be initialized from, π_o is the stationary policy defined over primitive actions followed during the lifetime of o, and $\beta_o : X \to [0,1]$ determines the probability that o will terminate while in a given state (Sutton et al., 1999). For each state $x \in X$, we denote the set of options that can be initialized from x by $\mathcal{O}_x = \{o \in \mathcal{O} \mid x \in \mathcal{I}_o\}$. The duration of an option is a random variable that depends on the state where the option was initialized and where the option terminates. For a state $x \in X$, an option $o \in \mathcal{O}_x$, and a subset of the state-space $Y \subseteq X$, $D^o_{x,Y}$ denotes the duration of executing option o from state x given that the option terminates in Y.

For an option $o = \langle \mathcal{I}_o, \pi_o, \beta_o \rangle$, we denote the probability that o is initialized from a state x and terminates in a subset of states $Y \subseteq X$ in exactly t timesteps by $P^o_t(Y|x)$ and the discounted termination state probability distribution of o by $\widetilde{P}^o(Y|x) = \sum_{t=1}^\infty \gamma^t P^o_t(Y|x)$. For a state-option pair (x,o), the discounted cumulative reward distribution during the option's execution is denoted by $\widetilde{R}(x,o)$.

The objective of planning in SMDPs is to derive a policy $\varphi : X \to \mathcal{O}$ from states to options that maximizes

$$V^\varphi(x) = \mathbb{E}\left[\widetilde{R}(x,\varphi(x))\right] + \int \widetilde{P}^{\varphi(x)}(y|x)V^\varphi(y)dy \ . \tag{5}$$

The Bellman operator for an SMDP is defined by

$$(\mathbb{T}V)(x) = \max_{o\in\mathcal{O}_x} \left(\mathbb{E}\left[\widetilde{R}(x,o)\right] + \int \widetilde{P}^o(y|x)V(y)dy \right) \ , \tag{6}$$

where $\mathbb{T}$ is defined over the set of options $\mathcal{O}$ instead of primitive actions A. The differences between (1) and (6) could potentially lead to widely different results when embedded in the FVI algorithm. However, in this paper, we will consider the case where $A \subset \mathcal{O}$, which ensures that $\mathcal{T}$ and $\mathbb{T}$ have the same fixed point $\mathcal{T}V^* = \mathbb{T}V^* = V^*$.

We introduce a generalization of FVI for the case where samples are generated by options (with primitive actions as a special case). The algorithm, Options Fitted Value Iteration (OFVI), takes as arguments positive integers N, L, K, $\mu \in M(X)$, an initial value function estimate $V_0 \in \mathcal{F}$, and a simulator $\mathbb{S}$. At each iteration k, N states $x_i \sim \mu$ for $i = 1,2,\ldots,N$ are sampled, and for each option $o \in \mathcal{O}_{x_i}$, L next states, rewards, and option durations $\langle y^o_{i,j}, r^o_{i,j}, \tau^o_{i,j} \rangle \sim \mathbb{S}(x_i,o)$ are sampled for $j = 1,2,\ldots,L$.

Then the update resulting from applying the Bellman operator to the previous iterate V_{k-1} is estimated by

$$\hat{V}_k(x_i) \leftarrow \max_{o \in \mathcal{O}_{x_i}} \frac{1}{L} \sum_{j=1}^{L} \left[r^o_{i,j} + \gamma^{\tau^o_{i,j}} V_{k-1}(y^o_{i,j}) \right] \quad , \quad (7)$$

and we obtain the best fit according to (3). In addition to returning a next state and reward, $\mathbb{S}$ also returns the number of timesteps that the option executed before terminating. This additional information is needed to compute (7). Otherwise, the differences between PFVI and OFVI are minor and it is natural to ask if OFVI has similar finite-sample and convergence behavior compared to PFVI.

Proposition 1. *For any $\varepsilon, \delta > 0$ and $K \geq 1$. Fix $p \geq 1$. Given a set of options $\mathcal{O}$ containing all primitive actions A, an initial state distribution $\rho \in M(X)$, a sampling distribution $\mu \in M(X)$, and $V_0 \in B(X, V_{\max})$, if A1(ρ, μ) holds, then there exist positive integers N and L such that when OFVI is executed, the policy φ_K returned by OFVI satisfies*

$$\begin{aligned} ||V^* - V^{\varphi_K}||_{p,\rho} &\leq \frac{2\gamma}{(1-\gamma)^2} C^{1/p}_{\rho,\mu} b_{p,\mu}(\mathbb{T}\mathcal{F}, \mathcal{F}) + \varepsilon \\ &+ \left(\gamma^{K+1}\right)^{1/p} \left(\frac{2\|V^* - V_0\|_\infty}{(1-\gamma)} \right) \end{aligned} \quad (8)$$

holds with probability at least $1 - \delta$.

A proof of Proposition 1 as well as sufficient values for N and L are given in the supplementary material. Inequality (8) bounds the suboptimality of the policy returned by OFVI, similar to inequality (4) from Munos & Szepesvári (2008). As long as the set of options $\mathcal{O}$ contains all primitive actions A, OFVI has performance in the worst case that is comparable to PFVI. The main differences between the bound in Proposition 1 and Munos & Szepesvári (2008, Theorem 2) is that the inherent Bellman error in Proposition 1 may be larger than the inherent Bellman error with only primitive actions, because backups computed by OFVI may span multiple timesteps resulting in more complex targets for the supervised learning algorithm to fit. However, the last terms characterizing the convergence rates of (4) and (8) are identical.

Proposition 1 implies that OFVI always converges approximately as fast as PFVI. However, the two algorithms may converge to different regions of the value function space due to the larger inherent Bellman error of OFVI. In the following section, we will investigate conditions where OFVI converges faster than PFVI.

4. Convergence Rate of OFVI

We are interested in analyzing the case where OFVI plans with an option set consisting of the set of primitive actions and a few additional temporally extended actions. In most cases, a temporally extended action can only be initialized from a subset of the state-space. The following defines the set of states that have access to temporally extended actions that follow approximately optimal policies from those states.

Definition 1. *Let $\mathcal{O}$ be the given set of options, $\alpha \geq 0$, and $d \geq 1$. The (α, d)-**omega set** $\omega_{\alpha,d}$ contains the states $x \in X$ such that there exists an option $o \in \mathcal{O}_x$ satisfying (1) the duration of executing o from x satisfies $\inf_{Y \subseteq X} \mathbb{E}\left[D^o_{x,Y}\right] \geq d$; and (2) o is α-optimal with respect to x (i.e. $Q^*(x, o) \geq V^*(x) - \alpha$).*

Temporally extended actions are only useful for planning if they are frequently encountered. The following assumption guarantees that, even if the temporally extended actions are sparsely scattered throughout the state-space, then they are not too difficult to reach from any state that we are likely to encounter starting from $x_0 \sim \rho$.

Assumption 2. $[A2(\alpha, d, \psi, \rho, j)]$ *Let $\alpha, \psi \geq 0$, $d, j \geq 1$, and $\rho \in M(X)$. For any m primitive policies $\pi_1, \pi_2, \ldots, \pi_m$, let $\nu = \rho P^{\pi_1} P^{\pi_2} \ldots P^{\pi_m}$. There exists an α-optimal policy $\hat{\pi}$ such that either (1) $\Pr_{x \sim \nu}[x \in \omega_{\alpha,d}] \geq 1 - \psi$; or (2) $\exists_{i \in \{1,2,\ldots,j\}} \Pr_{y \sim \eta_i}[y \in \omega_{\alpha,d}] \geq 1 - \psi$ where $\eta_i = \nu \left(P^{\hat{\pi}}\right)^i$ for $i = 1, 2, \ldots, j$.*

A2$(\alpha, d, \psi, \rho, j)$ assumes that, starting from $x \sim \rho$, at each timestep every possible trajectory either encounters a state in $\omega_{\alpha,d}$ with high probability $(1 - \psi)$, meaning that the agent almost always encounters states with temporally extended actions that are useful for planning, or from the agent's current state there exists a policy $\hat{\pi}$ that transitions to $\omega_{\alpha,d}$ with high probability in at most j timesteps. Under A2, useful temporally extended actions do not need to be at every state. They may be scattered sparsely throughout the state space. This assumption is weak since it can be made true for any MDP and set of options containing the primitive actions by tuning the parameter values. Furthermore, the agent does not need to know $\hat{\pi}$. It is sufficient that $\hat{\pi}$ exists.

Faster convergence depends critically on the optimism or pessimism of the sequence of iterates produced by OFVI. We say that an estimate $V \in B(X; V_{\max})$ of the optimal value function is optimistic if $V(x) > V^*(x)$ for all $x \in X$, and we say that V is pessimistic if $V(x) \leq V^*(x)$ for all $x \in X$. In fact, the SMDP Bellman operator $\mathbb{T}$ has a slower convergence rate than the MDP Bellman operator $\mathcal{T}$ when acting on entries of $V \in B(X; V_{\max})$ that are optimistic (Hauskrecht et al., 1998). This means that OFVI can only converge more quickly than PFVI when some of the iterates $\{V_k\}_{k=0}^K$ are pessimistic in at least part of the state-space. For standard value iteration this is not a problem because we can set the initial estimate V_0 to be pessimistic and the fact that $\mathbb{T}$ is monotonic and converges to V^* en-

sures that every iterate is also pessimistic. The situation for OFVI is more complicated because of the algorithm's fitting step. However, Theorem 1 (below) describes when we can expect OFVI to converge faster than PFVI provided that the first few iterates happen to be pessimistic with respect to V^*.

Theorem 1. *Let $\varepsilon, \delta > 0$, $\alpha, \psi, j \geq 0$, $K, p, d \geq 1$, $0 \leq Z \leq K$, and $\rho, \mu \in M(X)$. Suppose that $A1(\rho, \mu)$ and $A2(\alpha, d, \psi, \rho, j)$ hold, then if the first Z iterates $\{V_k\}_{k=0}^{Z}$ produced by the algorithm are pessimistic (i.e., $V_k(x) \leq V^*(x)$ for all $x \in X$), then there exist positive integers N and L such that when OFVI is executed, the policy φ_K returned by OFVI satisfies*

$$\begin{aligned} ||V^* - V^{\varphi_K}||_{p,\rho} \quad &\leq \tfrac{2\gamma}{(1-\gamma)^2} C_{\rho,\mu}^{1/p} \left(b_{p,\mu}(\mathbb{T}\mathcal{F}, \mathcal{F}) + \alpha\right) + \varepsilon \\ &+ \left(\gamma^{K+1+(1-\psi)(d-1)\lfloor Z/\bar{j} \rfloor}\right)^{1/p} \\ &\cdot \left(\tfrac{2\|V^* - V_0\|_\infty}{(1-\gamma)}\right) \end{aligned} \tag{9}$$

holds with probability at least $1 - \delta$, where $\bar{j} = j + 1$.

The proof of Theorem 1 is in the supplementary material. The OFVI algorithm runs exactly as we assumed for Proposition 1 using the same parameters and requires no special prior knowledge or preparation besides setting V_0 pessimisticly. For the results of the theorem to hold, at least the first Z iterates produced by the algorithm must be pessimistic. This condition is difficult to control in general, but it may be possible to control for specific applications and function approximation architectures, as we show below.

As with Proposition 1, the bound in Theorem 1 has three terms: (1) approximation error, (2) estimation error, and (3) convergence error. The main advantage of Theorem 1 can be seen in the third term, which characterizes the convergence rate of the algorithm. The algorithm converges at a rate of $\gamma^{K+1+(1-\psi)(d-1)\lfloor Z/\bar{j} \rfloor}$ rather than γ^{K+1} over K iterations. As $\psi \to 0$ and j decreases toward 0, OFVI can better exploit temporally extended actions to decrease the convergence error more quickly than PFVI. The stair-step nature of the convergence rate caused by $\lfloor Z/\bar{j} \rfloor$ is due to the possibility that all of the error in each iterate may concentrate on states outside of $\omega_{\alpha,d}$. In that case, it can take up to j iterations to propagate back the results of updates where temporally extended actions are used.

The main limitation of Theorem 1 is due to the first term, which controls the approximation error. This term is slightly worse than the approximation error in (8) due to our exploitation of α-optimal options and policy $\hat{\pi}$. However, this does not imply that the algorithm converges to a worse solution than (8). It reflects the fact that when the temporally extended actions are suboptimal, convergence will be rapid up to a point. Once the algorithm achieves an accurate iterate, the convergence rate may slow because the SMDP Bellman operator cannot improve the current estimate further with temporally extended actions.

5. Experiments & Results

We compared PFVI and OFVI in two different tasks: (1) the optimal replacement problem considered in Munos & Szepesvári (2008), and a cyclic eight-commodity inventory management task. In both experiments, we see that options can improve convergence rates of FVI, but only when V_0 is pessimistic with respect to V^*.

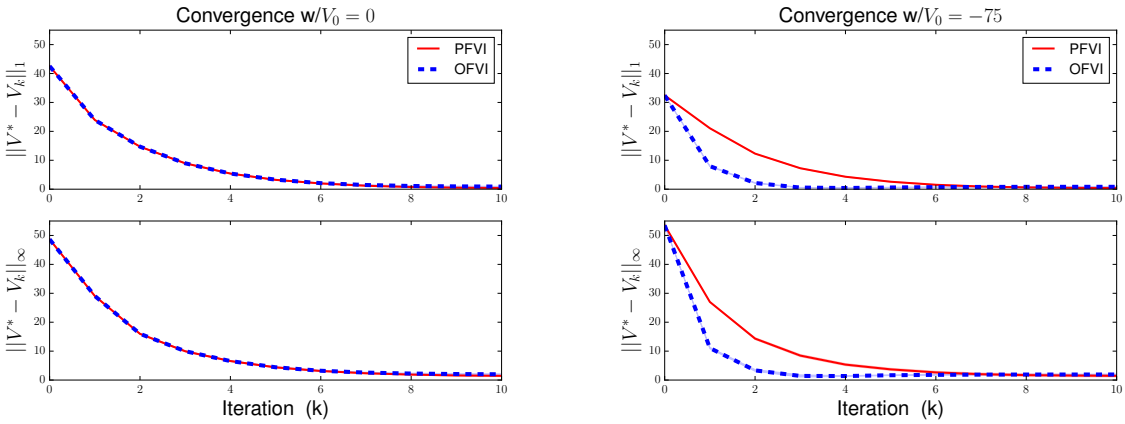
5.1. Optimal Replacement Task

In the optimal replacement problem, the agent selects from one of two actions K and R, whether to maintain a product (action K) at a maintenance cost $c(x)$ that depends on the product's condition x or replace (action R) the product with a new one for a fixed cost C. This problem is easy to visualize because it has only a single dimension, and the optimal value function and optimal policy can be derived in closed form (Munos & Szepesvári, 2008) so that we can compare PFVI and OFVI directly with the optimal policy. We used parameter values $\gamma = 0.6$, $\beta = 0.5$, $C = 30$ and $c(x) = 4x$ (identical to those used by Munos & Szepesvári (2008)) where β is the inverse of the mean of an exponential distribution driving the transition dynamics of the task. Similar to Munos & Szepesvári (2008), we used polynomials to approximate the value function. All results presented here used fourth degree polynomials. The optimal policy keeps the product up to a point $\bar{x}$ and replaces the product once the state equals or exceeds $\bar{x}$.

For the OFVI condition, we introduced a single option that keeps the product up to a point $\tilde{x} = \bar{x} + \Delta$ and terminates once the state equals or exceeds $\tilde{x}$. By modifying Δ, we controlled the optimality of the given option. As predicted by our analysis, adjusting Δ away from 0 (i.e., increasing the suboptimality of the option), resulted in slower convergence when the initial value function was pessimistic. For an optimistic initial value function, the behavior of PFVI and OFVI was almost identical.

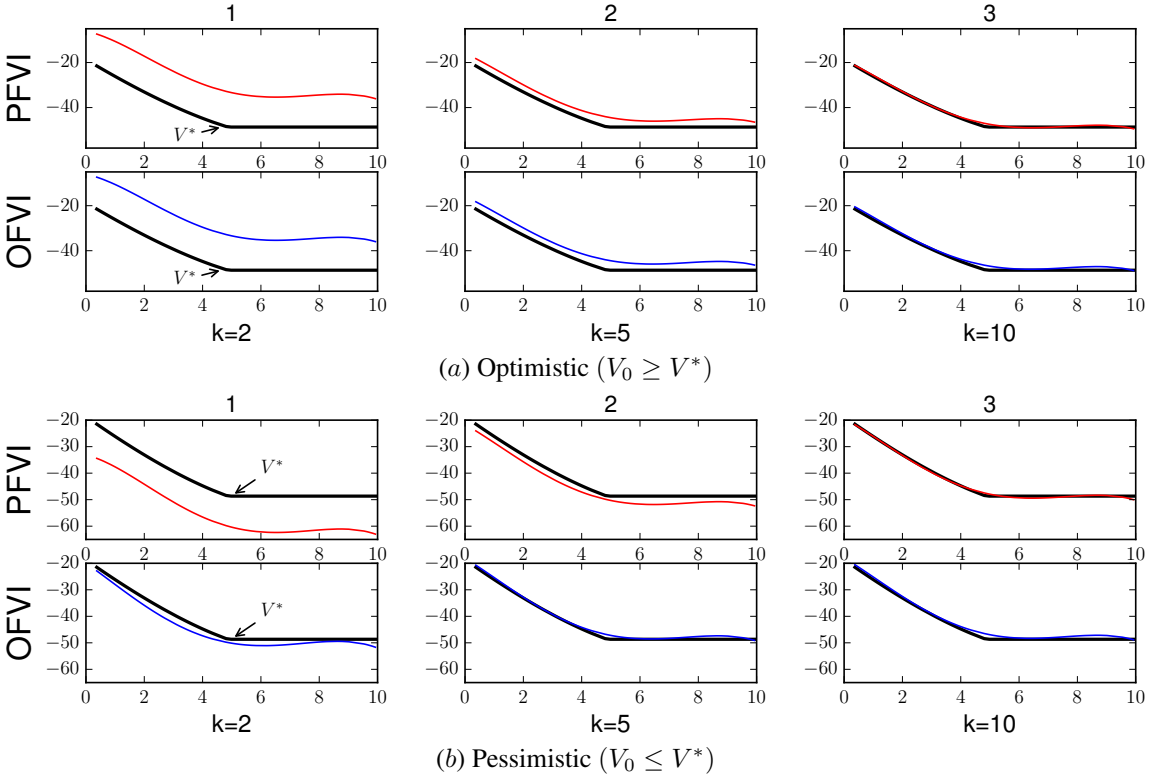
Figure 1a shows the average convergence rates of PFVI and OFVI (with $\Delta = 0$) when the initial value function estimate is optimistic for both max-norm and L_1-norm error. In both cases, the value functions converge at almost identical rates, as predicted by our analysis. Figure 1b shows the average convergence rates of PFVI and OFVI when the initial value function estimate is pessimistic. With a pessimistic initial value function, OFVI converges significantly faster than PFVI as predicted by our analysis.

Figure 2 compares the average iterates V_k of OFVI to PFVI for $k = 2, 5,$ and 10 with optimistic (Figure 2a) and pessimistic (Figure 2b) initial value function estimates. The



(a) Optimistic ($V_0 \geq V^*$) (b) Pessimistic ($V_0 \leq V^*$)

Figure 1. Optimal Replacement Task: Convergence rates of PFVI and OFVI averaged over 100 trials (std. deviations are too small to visualize). (a) When the initial value function estimate V_0 is optimistic, there is no difference between the convergence rates of PFVI and OFVI. (b) However, when V_0 is pessimistic, OFVI converges faster than PFVI.



(a) Optimistic ($V_0 \geq V^*$)

(b) Pessimistic ($V_0 \leq V^*$)

Figure 2. Optimal Replacement Task: Average iterates V_k ($k = 2, 5,$ and 10) for PFVI and OFVI. (a) Columns 1, 2, and 3 show that the convergence rate of OFVI and PFVI are qualitatively similar when the initial value function is optimistic. (b) When the initial value function is pessimistic, OFVI's value function estimate after $k = 2$ iterations (column 1) is qualitatively similar to PFVI's value function estimate after $k = 5$ iterations (column 2).

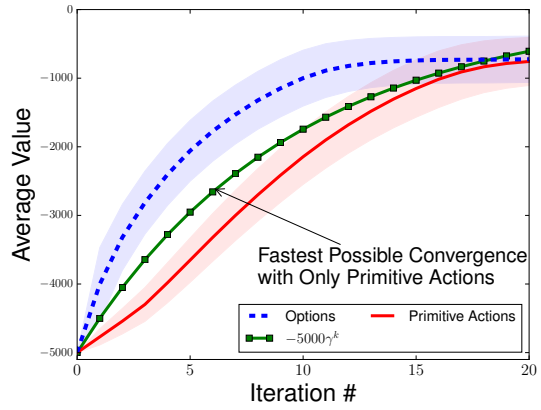


Figure 3. Inventory Management Task: Average values predicted by pessimistic iterates for PFVI and OFVI (shaded regions denote ± 1 std. deviation).

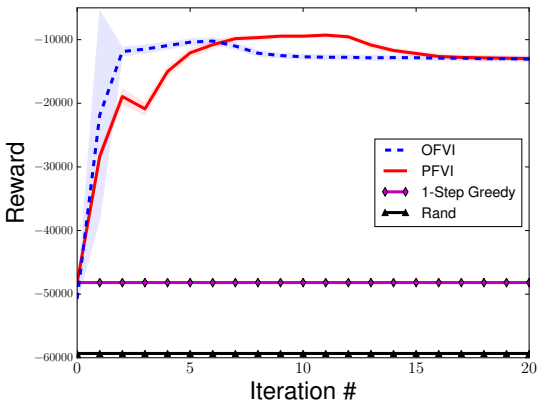


Figure 4. Inventory Management Task: Cumulative reward of policies derived from PFVI and OFVI after each iteration compared to uniform random (Rand) and 1-step greedy policies (shaded regions denote ± 1 std. deviation).

solid black line depicts the optimal value function V^*. With an optimistic initial value function the behavior of PFVI and OFVI is qualitatively identical. However, with a pessimistic initial value function, OFVI's second iterate is qualitatively similar to PFVI's fifth iterate.

5.2. Cyclic Inventory Management Task

In a basic inventory management setting, an agent controls the order policy for a single commodity (Scarf, 1959). Each round, the demand for the commodity is revealed (sampled from a distribution) and subtracted from the agent's inventory. The agent decides the quantity to order, and the order is filled immediately. If the agent did not have sufficient supply to meet the demand it receives a high penalty (i.e., large negative reward). On the other hand, ordering commodities and storing them are also penalized (i.e., given negative rewards). The objective is to find the policy that balances these penalties over time.

Cyclic inventory management problems are further complicated because the demand distribution changes after each round, but the distributions repeat after a finite number of rounds (Sethi & Cheng, 1997).

We introduce an eight-commodity, cyclic inventory management problem with finite storage. The demand distributions cycle every 12 months and there are 24 rounds per year. The agent must manage eight different commodities that are stored together in a finite warehouse. Ordering too much of quantity i means that there is less room in the warehouse for quantity $j \neq i$. Thus the agent must work out complex trade-offs that depend on both the current inventory levels and the time of year. The details of the task and exact parameters used in our experiments are described in the supplementary material.

The state-space was described by $\langle \tau, x \rangle$ where $\tau \in \{1, 2, \ldots, 24\}$ denotes the period in the cycle and x is a vector determining the quantity of each commodity stored in the warehouse. To approximate the value function, we partitioned the state-space by the 24 periods in the cycle. Thus, each iterate was constructed from 24 independent function approximators. Because of the high dimensionality of the inventory space, we needed a function approximation architecture with good generalization properties. After experimenting with various architectures, we found that linear approximations with a fixed grid of one-dimensional radial basis functions generalized well with limited samples. Cross-validation was used to select grid density and basis widths.

For the OFVI condition, we created options based on the intuition that good inventory management policies order commodities in large quantities and make zero orders on as many rounds as possible to avoid the base ordering penalty. We defined options that always place zero orders and terminate once the inventory level of a specific commodity drops below a threshold. Options were added for twenty different threshold levels for each commodity.

This problem is difficult for two reasons. First, the agent must manage eight different commodities with limited, shared storage. Making a large order of one commodity reduces the space available for other commodities. Second, the demand distributions are cyclic requiring adaptation to the time of year. The agent must plan ahead stockpiling commodities when demand is low. However, if the agent fills its warehouse, it will not be able to adapt to unexpected situations that arise due to the problem's stochastic demands.

Figure 3 shows average (over 5,000 sampled states) pre-

dicted values for PFVI and OFVI, when V_0 is pessimistic. The average predicted value increases more rapidly for OFVI than PFVI. The curve marked with squares in Figure 3 depicts the fastest possible convergence rate with only primitive actions when V_0 is pessimistic. On most iterations, this curve falls well below the convergence rate of OFVI until OFVI seems to have approximately converged.

Figure 4 shows cumulative reward received by policies derived from the iterates of PFVI and OFVI when V_0 is pessimistic. Cumulative rewards were recorded over 100 timesteps starting from a state with zero inventory. Policies derived from PFVI and OFVI outperform random and 1-step greedy policies after a single iteration. Policies derived from OFVI converge to a good solution with fewer iterations than PFVI. The decrease in performance in policies derived from PFVI and OFVI in later iterations is due to approximation error (the fact that our function approximation architecture does not exactly fit the points generated by Bellman backups explained by the first term on the right hand side of (4),(8), and (9)). When V_0 is optimistic, policies derived from PFVI and OFVI achieve similar cumulative reward at each iteration (not shown).

6. Discussion

We demonstrated two different tasks where augmenting the primitive actions with temporally extended actions leads to faster convergence. As our theoretical analysis predicted, when V_0 was pessimistic, the additional temporally extended actions helped OFVI to converge faster than PFVI. However, adding additional options increases the computational and sample complexity of each iteration of OFVI. Thus, randomly generating hundreds of options will probably not lead to overall improvement in the computational complexity of AVI. However, adding a few options does not significantly increase the cost of each iteration.

A natural extension of this work is to consider automatically generating options that speed up planning. Many previous works have looked at generating options (McGovern & Barto, 2001; Stolle & Precup, 2002; Mannor et al., 2004; Silver & Ciosek, 2012). What is missing from the literature are theoretical analyses that enables us to evaluate and compare different strategies for generating options.

Options may have other benefits for planning besides improving the convergence rate. For example, options may enable a planning algorithm to "skip over" regions of the state-space with highly complex dynamics without impacting the quality of the planned policy. In partially observable environments, options may be exploited to decrease uncertainty about the hidden state by "skipping over" regions of the state-space where there is large observation variance, or "testing" hypotheses about the hidden state (Mann et al., 2013). Options may also play an important role in robust optimization, where the dynamics of temporally extended actions are known with greater certainty than the dynamics of a sequence of primitive actions.

Acknowledgements

The research leading to these results has received funding from the European Research Counsel under the European Unions Seventh Framework Program (FP7/2007-2013) / ERC Grant Agreement No 306638.

References

Hauskrecht, Milos, Meuleau, Nicolas, Kaelbling, Leslie Pack, Dean, Thomas, and Boutilier, Craig. Hierarchical solution of markov decision processes using macro-actions. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp. 220–229, 1998.

He, Ruijie, Brunskill, Emma, and Roy, Nicholas. Efficient planning under uncertainty with macro-actions. *Journal of Artificial Intelligence Research*, 40:523–570, 2011.

Mann, Timothy A., Park, Yunjung, Jeong, Sungmoon, Lee, Minho, and Choe, Yoonsuck. Autonomous and interactive improvement of binocular visual depth estimation through sensorimotor interaction. *Autonomous Mental Development, IEEE Transactions on*, 5(1):74–84, 2013. ISSN 1943-0604. doi: 10.1109/TAMD.2012.2216524.

Mannor, Shie, Menache, Ishai, Hoze, Amit, and Klein, Uri. Dynamic abstraction in reinforcement learning via clustering. In *Proceedings of the Twenty-First International Conference on Machine learning*, ICML '04, pp. 71–, New York, NY, USA, 2004. ACM. ISBN 1-58113-838-5. doi: 10.1145/1015330.1015355. URL http://doi.acm.org/10.1145/1015330.1015355.

McGovern, Amy and Barto, Andrew G. Automatic Discovery of Subgoals in Reinforcement Learning using Diverse Density. In *Proceedings of the 18th International Conference on Machine Learning*, pp. 361 – 368, San Fransisco, USA, 2001.

Munos, Rémi. Error bounds for approximate value iteration. In *Proceedings of the National Conference on Artificial Intelligence*, 2005.

Munos, Rémi and Szepesvári, Csaba. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9:815–857, 2008.

Precup, Doina, Sutton, Richard S, and Singh, Satinder. Theoretical results on reinforcement learning with tem-

porally abstract options. In *Machine Learning: ECML-98*, pp. 382–393. Springer, 1998.

Scarf, Herbert. The optimality of (s,s) policies in the dynamic inventory problem. Technical Report NR-047-019, Office of Naval Research, April 1959.

Sethi, Suresh P. and Cheng, Feng. Optoptimal of (s,s) policies in inventory models with markovian demand. *Operations Research*, 45(6):931–939, 1997.

Silver, David and Ciosek, Kamil. Compositional Planning Using Optimal Option Models. In *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, 2012.

Stolle, Martin and Precup, Doina. Learning options in reinforcement learning. In *Abstraction, Reformulation, and Approximation*, pp. 212–223. Springer, 2002.

Sutton, Richard S, Precup, Doina, and Singh, Satinder. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1):181–211, August 1999.

Theocharous, Georgios and Kaelbling, Leslie P. Approximate planning in pomdps with macro-actions. In *Advances in Neural Information Processing Systems*, pp. None, 2003.